

# Projektdokumentation

---

Tulip  
Blum, Manojlovic, Naray, Zurkinden, Jashari

Datum	Version	Änderung	Autor
11.05.2022	0.0.1	Erste Version	Naray
18.05.2022	0.0.2	Arbeitspakete ergänzt	Naray
25.05.2022	0.0.3	HTML und CSS für Unterseiten fertiggestellt. (Vorlage Struktur)	Naray
	1.0.0	Finale Version	

## 1. Informieren

### 1.1 Ihr Projekt

---

Unser Projekt wird eine Website sein, die **den Einstieg in das Programmieren erleichtern** soll. Es soll Unterseiten mit oberflächlicher und vereinfachter Erklärung zu **Themen bezüglich OOP** haben. Ein anderer Teil soll eine Liste an **Projektideen** werden, welche man mit dem angeschafften Wissen realisieren kann.

### 1.2 Quellen

---

[Listen Sie hier explizit alle Quellen, die Sie zu benutzen planen, um sich in das Projekt einzuarbeiten. Aktualisieren Sie diesen Teil, wenn Sie die Quellen bearbeitet haben oder weitere Quellen hinzugekommen sind.]

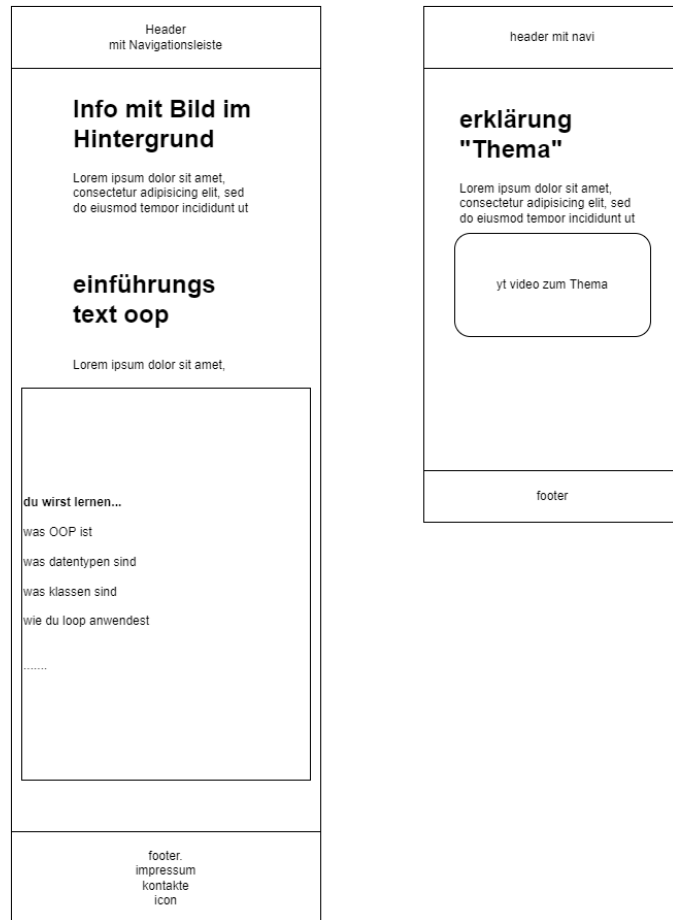
### 1.3 Anforderungen

Nummer	Muss / Kann?	Funktional? Qualität? Rand?	Beschreibung
1	Muss	Funktional	Die Webseite muss funktionierende Verlinkungen haben
2	Muss	Funktional	Header mit Logo und Navigationsleiste
3	Muss	Funktional	Footer mit Kontakt und anderen Verlinkungen (z. B. Impressum)
4	Muss	Funktional	Ein paar Code-Embeds
5	Muss	Funktional	Eingebettete Youtube-Videos zu den Themen
6	Muss	Qualität	Hover-Effekte für Buttons oder Textabschnitte
7	Kann	Qualität	Lexikonseite für Fachbegriffe
8	Kann	Rand	Alle Fachbegriffe haben eine Verlinkung zur Lexikonseite mit der Definition
9	Muss	Qualität	Seite mit Projektideen und deren Erklärungen
10	Kann	Qualität	Englische Seite
11	Muss	Funktional	Eine organisierte Folderstruktur ist vorhanden
12	Muss	Funktional	Es hat Farbvariablen für CSS-Farben. (für leichtere Farbanpassung)
13	Muss	Funktional	Die Container überlappen nicht
14	Muss	Qualität	Jede Seite hat ein gemeinsames Schema
15	Muss	Qualität	Alle Seiten sind responsive für die Desktopversion

## 1.4 Diagramme

---

Das ist das Mockup für unsere Homepage und eine Vorlage für Unterseiten.



## 1.5 Testfälle

---

[Erstellen Sie zu jeder Muss-Anforderung mindestens einen Testfall.]

Nummer	Vorbereitung	Eingabe	Erwartete Ausgabe

\* Die Nummer hat das Format N.m, wobei N die Nummer der Anforderung ist, die mit dem Test abgedeckt wird, und m von 1 an fortlaufend durchnummeriert wird.

## 1.6 github-Ablage

---

<https://github.com/mnaray/CrashcourseOOP>

## 2. Planen

Nummer	Frist	Beschreibung	Zeit (geplant)
11.1	18.05	Folderstruktur wird erstellt und die Konventionen in der Gruppe geklärt	45
12.1	18.05	Farbvariablen für CSS erstellen	15
2.1	18.05	Header und Footer zusammen im Unterricht erstellen.	90
4.1	18.05	Recherche betreiben über Code-Embed in HTML	15
4.2	18.05	Vorlage für Code-Embeds erstellen	15
5.1	18.05	Recherche betreiben über Youtube-Embeds in HTML	15
5.2	18.05	Vorlage für Youtube-Embeds	15
13.1	25.05	HTML Vorlage für die Unterseiten schreiben (noch mit lorem ipsum)	90
13.2	25.05	Die Seitenstruktur für Mobile wird in CSS geschrieben	180
13.3	25.05	Unterseiten werden alle individualisiert	180
6.1	01.06	Hoveranimationen in CSS-Klassen schreiben	45
6.2	01.06	Klassen den Buttons zuweisen	45
14.1	01.06	Farbschema für jede Seite implementieren	90
1.1	01.06	Testprotokoll wird geschrieben	45
15.1	08.06	Responsive für Desktopversion erstellen	180-225
-	15.06	Debuggen und Korrekturen/Verbesserungen vornehmen	180
-	15.06	Ergänzen der Dokumentation	45
-	15.06	Besprechung vom momentanen Zustand des Projektes (evtl. Prozentbeträge anpassen)	45
TOTAL:			Sitzungen × Lektionen × Gruppenmitglieder

## 3. Entscheiden

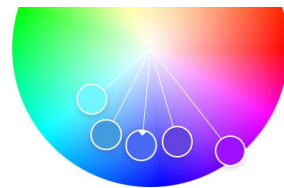
Wir hatten die Option eine Bilinguale Seite zu schreiben, jedoch haben wir uns entschieden diese Option als "kann" einzutragen. Wir werden die englische Seite implementieren, falls wir noch Zeit übrighaben.

## Bewertung:

Wir haben uns für eine gleichmässige Aufteilung der Punkte entschieden. Alle bekommen 20%, wenn alle mitarbeiten. Ansonsten würden wir die Beträge im Fall der Fälle anpassen, damit es fair ist.

## Farbschema:

zentral  
zentral teilen  
zentral doppelt teilen  
  
engesetzt  
ungen  
definiert



**Font:**

*Serif:* <https://fonts.google.com/specimen/IBM+Plex+Serif?category=Serif>

*Sans Serif:* <https://fonts.google.com/specimen/Ubuntu?category=Sans+Serif>

*Code:* [https://fonts.google.com/specimen/Roboto?preview.text=programming&preview.text\\_type=custom](https://fonts.google.com/specimen/Roboto?preview.text=programming&preview.text_type=custom)

#### 4. Realisieren

Nummer	Datum	Beschreibung	Zeit (geplant)	Zeit (effektiv)
11.1	18.05	Folderstruktur wird erstellt und die Konventionen in der Gruppe geklärt	45	60
12.1	18.05	Farbvariablen für CSS erstellen	15	15
2.1	18.05	Header und Footer zusammen im Unterricht erstellen.	90	90
4.1	18.05	Recherche betreiben über Code-Embed in HTML	15	10
4.2	18.05	Vorlage für Code-Embeds erstellen	15	5
5.1	18.05	Recherche betreiben über Youtube-Embeds in HTML	15	10
5.2	18.05	Vorlage für Youtube-Embeds	15	5
13.1	25.05	HTML Vorlage für die Unterseiten schreiben (noch mit lorem ipsum)	90	75
13.2	25.05	Die Seitenstruktur für Mobile wird in CSS geschrieben	180	60

[Übernehmen Sie Ihre Planung aus 2., und tragen Sie nach, wie lang Sie effektiv zur Bearbeitung der jeweiligen Arbeitspakete benötigt haben.]



## 5. Kontrollieren

### 5.1 Testprotokoll

Nummer	Datum	Resultat	Durchgeführt
			[Nachname]

[Vergessen Sie Ihr Fazit nicht!]

### 5.2 Exploratives Testen

Nummer	Vorbereitung	Eingabe	Erwartete Ausgabe	Effektive Ausgabe
A*	Applikation gestartet, Benutzer wird nach Name gefragt	HansÜeli	Willkommen, HansÜeli	Programm bricht mit string parse-Error ab
...				

\* Verwenden Sie hier Großbuchstaben (A, B, ...), um die einzelnen Befunde zu nummerieren.

## 6. Auswerten

[Listen Sie hier je mindestens einen Punkt, der gut gelaufen ist, und einen Punkt, der schlecht gelaufen ist – mit diesen starten Sie dann in Ihren Portfolio-Eintrag.]